```
PPPPPPPPPPPP     RRRRRRRRRRRR     TTTTTTTTTTTTTTT     SSSSSSSSSSS  MMM         MMM  BBBBBBBBBBBB
PPPPPPPPPPPPP    RRRRRRRRRRRRR    TTTTTTTTTTTTTTT     SSSSSSSSSSSS MMM         MMM  BBBBBBBBBBBBB
PPPPPPPPPPPPP    RRRRRRRRRRRRR    TTTTTTTTTTTTTTT     SSSSSSSSSSSS MMM         MMM  BBBBBBBBBBBBB
PPP          PPP RRR          RRR       TTT          SSS           MMMMMMM   MMMMMMM  BBB          BBB
PPP          PPP RRR          RRR       TTT          SSS           MMMMMMM   MMMMMMM  BBB          BBB
PPP          PPP RRR          RRR       TTT          SSS           MMMMMMM   MMMMMMM  BBB          BBB
PPP          PPP RRR          RRR       TTT          SSS           MMM MMM MMM MMM  BBB          BBB
PPP          PPP RRR          RRR       TTT          SSS           MMM MMM MMM MMM  BBB          BBB
PPP          PPP RRR          RRR       TTT          SSS           MMM MMM MMM MMM  BBB          BBB
PPPPPPPPPPPPP    RRRRRRRRRRRRR          TTT           SSSSSSSSS     MMM     MMM MMM  BBBBBBBBBBBB
PPPPPPPPPPPPP    RRRRRRRRRRRRR          TTT           SSSSSSSSS     MMM     MMM MMM  BBBBBBBBBBBB
PPPPPPPPPPPPP    RRRRRRRRRRRRR          TTT           SSSSSSSSS     MMM     MMM MMM  BBBBBBBBBBBB
PPP              RRR   RRR              TTT                   SSS  MMM     MMM MMM  BBB          BBB
PPP              RRR   RRR              TTT                   SSS  MMM     MMM MMM  BBB          BBB
PPP              RRR   RRR              TTT                   SSS  MMM     MMM MMM  BBB          BBB
PPP              RRR       RRR          TTT                   SSS  MMM     MMM MMM  BBB          BBB
PPP              RRR       RRR          TTT                   SSS  MMM     MMM MMM  BBB          BBB
PPP              RRR       RRR          TTT                   SSS  MMM     MMM MMM  BBB          BBB
PPP              RRR           RRR      TTT          SSSSSSSSSSS  MMM     MMM MMM  BBBBBBBBBBBB
PPP              RRR           RRR      TTT          SSSSSSSSSSS  MMM     MMM MMM  BBBBBBBBBBBB
PPP              RRR           RRR      TTT          SSSSSSSSSSS  MMM     MMM MMM  BBBBBBBBBBBB
```

```
  SSSSSSSS   MM       MM   BBBBBBBB    RRRRRRRR    EEEEEEEEEE    QQQQQQ
  SSSSSSSS   MM       MM   BBBBBBBB    RRRRRRRR    EEEEEEEEEE    QQQQQQ
SS           MMMM   MMMM   BB      BB  RR      RR  EE           QQ      QQ
SS           MMMM   MMMM   BB      BB  RR      RR  EE           QQ      QQ
SS           MM  MM  MM   BB      BB  RR      RR  EE           QQ      QQ
SS           MM  MM  MM   BB      BB  RR      RR  EE           QQ      QQ
  SSSSSS     MM      MM   BBBBBBBB    RRRRRRRR    EEEEEEEE     QQ      QQ
  SSSSSS     MM      MM   BBBBBBBB    RRRRRRRR    EEEEEEEE     QQ      QQ
        SS   MM      MM   BB      BB  RR  RR      EE           QQ  QQ  QQ
        SS   MM      MM   BB      BB  RR  RR      EE           QQ  QQ  QQ
        SS   MM      MM   BB      BB  RR    RR    EE           QQ    QQ       ....
        SS   MM      MM   BB      BB  RR    RR    EE           QQ    QQ       ....
SSSSSSSS     MM      MM   BBBBBBBB    RR      RR  EEEEEEEEEE    QQQQ  QQ     ....
SSSSSSSS     MM      MM   BBBBBBBB    RR      RR  EEEEEEEEEE    QQQQ  QQ     ....

RRRRRRRR    EEEEEEEEEE    QQQQQQ
RRRRRRRR    EEEEEEEEEE    QQQQQQ
RR      RR  EE           QQ      QQ
RR      RR  EE           QQ      QQ
RR      RR  EE           QQ      QQ
RR      RR  EE           QQ      QQ
RRRRRRRR    EEEEEEEE     QQ      QQ
RRRRRRRR    EEEEEEEE     QQ      QQ
RR  RR      EE           QQ  QQ  QQ
RR  RR      EE           QQ  QQ  QQ
RR    RR    EE           QQ    QQ
RR    RR    EE           QQ    QQ
RR      RR  EEEEEEEEEE    QQQQ  QQ
RR      RR  EEEEEEEEEE    QQQQ  QQ
```

```
!       Require file for print symbiont facility
!
! Version:      'V04-000'
!
!****************************************************************************
!*                                                                          *
!*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
!*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
!*  ALL RIGHTS RESERVED.                                                    *
!*                                                                          *
!*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
!*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE          *
!*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER     *
!*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
!*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY       *
!*  TRANSFERRED.                                                            *
!*                                                                          *
!*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE    *
!*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT        *
!*  CORPORATION.                                                            *
!*                                                                          *
!*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS     *
!*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
!*                                                                          *
!*                                                                          *
!****************************************************************************
!
!++
! FACILITY:
!       Symbiont Services.
!
! ABSTRACT:
!       Macro and literal definitions for symbionts.
!--
!
! AUTHOR: G. Robert, CREATION DATE: 01-May-1983
!
! MODIFIED BY:
!
!       3B-007  RRB3007         Rowland R. Bradley      20-Jul-1984
!               Change PSM$_ messages NOMOREITEMS, INVSTRLEV, and
!               INVSTMNBR to SMB$_NOMOREITEMS, SMB$_INVSTRLEV, and
!               SMB$_INSTRMNBR, respectively.  Delete messages
!               PSM$_INVSCB, PSM$_REQNOTSUP, PSM$_INVREQCOD.
!
!       3B-006  GRR3006         Gregory R. Robert       29-Apr-1984
!               Added PSM$_FLUSH
!
!       3B-005  RRB0014         Rowland R. Bradley      27-Apr-1984
!               Remove the Task_Flag macro.
!
!       3B-004  RRB0013         Rowland R. Bradley      21-Feb-1984
!               Add new STRUCTURE definition for PAGE.
```

```
!       3B-003  GRR3004         Gregory R. Robert       23-Aug-1983
!               Bugfixes, page_setup_modules, form_setup_modules,
!               sheet_feed, symbiont initiated pause_task and stop_stream,
!               hangup code, read and write item services
!
!       3B-003  GRR3003         Gregory R. Robert       03-Aug-1983
!               Fixes for new design.
!
!       3B-002  GRR3002         Gregory R. Robert       29-Jul-1983
!               Added several macros to access symbiont tables.
!
!       3B-001  GRR3001         Gregory R. Robert       23-Jun-1983
!               Fixed offset_table macro, added item_present macro,
!               fixed some bugs, added several literals.
!
!       3B-000  GRR3000         Gregory R. Robert       27-May-1983
!               Original version.
!**

! Define program section standard names and attributes
!
PSECT
        CODE    = CODE,
        PLIT    = CODE,
        OWN     = DATA,
        GLOBAL  = DATA
        ;


!Check that structure id's have a common byteoffset
!
$ASSUME ($BYTEOFFSET (IOB_L_STRUCTURE), EQL, $BYTEOFFSET (PSM$L_STRUCTURE))

! Check that a quadword of item code flags is adequate
!
$ASSUME (SMBMSG$K_MAX_ITEM_CODE - 1, LSS, 64)

! Delcare useful builtin Bliss functions
!
BUILTIN
        CALLG,
        FFS,
        INSQUE,
        LOCC,
        MOVC3,
        MOVC5,
        MOVTUC,
        REMQUE,
        TESTBITCC,
        TESTBITCS,
        TESTBITSC,
        TESTBITSS
        ;

! Declare special linkages
```

```
!
LINKAGE
        ANALYZE_LINKAGE = JSB (REGISTER=0; REGISTER=0, REGISTER=1),

!*!
!*! CAN'T USE THIS UNTIL UNDERSTAND HOW TO PRESERVE R2-R4 INCLUSIVE
!*!
        FREE_LINKAGE    = JSB (REGISTER=0)
        ;

! Declare common external routines
!
EXTERNAL ROUTINE
        BAS$EDIT,

        LBR$CLOSE,
        LBR$GET_RECORD,
        LBR$INI_CONTROL,
        LBR$LOOKUP_KEY,
        LBR$OPEN,
        LBR$RET_RMSSTV,
        LBR$SET_LOCATE,

        LIB$TRIM_FILESPEC,
        LIB$GET_VM,
        LIB$FREE_VM,

        STR$ANALYZE_SDESC,
        STR$ANALYZE_SDESC_R1 : ANALYZE_LINKAGE,
        STR$APPEND,
        STR$CONCAT,
        STR$COPY_DX,
        STR$COPY_R,
        STR$FREE1_DX,
        STR$FREE1_DX_R4      : FREE_LINKAGE,
        STR$GET1_DX,
        STR$LEFT,
        STR$PREFIX,
        STR$RIGHT
        ;

EXTERNAL LITERAL
        PSM$S_HANGUP_DISPATCH_ENTRY
        ;

! Private messages
!
EXTERNAL LITERAL
        PSM$_BUFFEROVF,
        PSM$_EOF,
        PSM$_ESCAPE,
        PSM$_FLUSH,
        PSM$_FUNNOTSUP,
        PSM$_INVITMCOD,
        PSM$_INVVMSOSC,
```

```
        PSM$_MODNOTFND,
        PSM$_NEWPAGE,
        PSM$_NOFILEID,
        PSM$_OSCTOOLON,
        PSM$_PENDING,
        PSM$_SUSPEND,
        PSM$_TOOMANYLEV,
        SMB$_INVSTMNBR,
        SMB$_INVSTRLEV,
        SMB$_NOMOREITEMS
        ;

! Shared messages
!
$SHR_MSGDEF (PSM, PSM$K_FACILITY, LOCAL,
        (BADLOGICPC, SEVERE),   ! - logic error with PC value
        (CLOSEIN, ERROR),       ! - unable to close input
        (OPENIN, ERROR),        ! - unable to open or connect to input
        (READERR, ERROR),       ! - error reading
        (WRITEERR, ERROR)       ! - error writing
        );


! Define structures useful for accessing parameters passed by reference
!
STRUCTURE
    $BYTE               [] = $BYTE <0,08,0>,
    $SIGNED_BYTE        [] = $SIGNED_BYTE <0,08,1>,

    $WORD               [] = $WORD <0,16,0>,
    $SIGNED_WORD        [] = $SIGNED_WORD <0,16,1>,

    $LONGWORD           [] = $LONGWORD <0,32,0>,
    $SIGNED_LONGWORD    [] = $SIGNED_LONGWORD <0,32,1>
    ;
!
! Define structures useful for referencing the "page" of information
!
STRUCTURE
    PAGE_ARRAY[I, J, K; N, M, UNITS=1] =        ! default is byte referencing
        [M * N * UNITS]
        (PAGE_ARRAY + (J * K + I) * UNITS)<0,8,0>
    ;
!
! Message Item Table (MIT) and Service Routine (SRV) table building macros
!

MACRO
        MIT_PRESET_ [TAG, RESET, TYPE, ITEM] =
            [%NAME ('SMBMSG$K_', ITEM), MIT_B_TYPE]   = %NAME ('MIT_K_', TYPE),
            [%NAME ('SMBMSG$K_', ITEM), MIT_V_RESET]  = %IF %NULL (RESET) %THEN 1 %ELSE RESET %FI,
            [%NAME ('SMBMSG$K_', ITEM), MIT_W_OFFSET] =
                $BYTEOFFSET (%NAME ('PSM$', TAG, '_', ITEM))
        % ;

MACRO
```

```
    SRV_PRESET_ [SERVICE, USER, TYPE] =
        [%NAME ('PSM$K_', SERVICE), SRV_A_SERVICE] =
            %NAME ('PSM$', SERVICE),
        [%NAME ('PSM$K_', SERVICE), SRV_V_USER_ALLOWED] =
            %IF %NULL (USER) %THEN NO_USER
            %ELSE USER %FI,
        [%NAME ('PSM$K_', SERVICE), SRV_B_SERVICE_TYPE] =
            %IF %NULL (TYPE) %THEN SRV_K_GENERAL_SERVICE
            %ELSE %NAME ('SRV_K_', TYPE, '_SERVICE') %FI
        % ;


! General purpose macros
!
MACRO
        ACC_DATA_ (ITEM) =
            $BBLOCK [SCB[PSM$T_ACCOUNTING_AREA], %NAME ('SMBMSG$L_', ITEM)]
        %,

        BLINK_ (QUEUE_HEADER) =
            VECTOR [QUEUE_HEADER, 1]
        %,

        CLEAR_QUAD_ (QUAD) =
            BEGIN
            VECTOR [QUAD, 0] = 0;
            VECTOR [QUAD, 1] = 0;
            END
        %,

    CLEAR_STRING_ (DESC) =
        BEGIN
        IF .DESC_CLASS_ (DESC) LEQU DSC$K_CLASS_S   ! 0 or 1
        THEN
            INIT_DYN_DESC_ (DESC)
        ELSE
            IF .DESC_SIZE_ (DESC) NEQ 0
            THEN
                STR$FREE1_DX (DESC)
        END
    %,

    CODEERR_ =
        SIGNAL_STOP (PSM$_BADLOGICPC, 1)    ! PC implied 3rd arg
    %,

    COPY_DX_DX_ (FROM_DESC, TO_DESC) =
        SIGNAL_IF_ERROR_ (STR$COPY_DX (TO_DESC, FROM_DESC))
    %,

    COPY_R_DX_ (FROM_SIZE, FROM_ADDRESS, TO_DESC) =
        SIGNAL_IF_ERROR_ (STR$COPY_R (TO_DESC, FROM_SIZE, FROM_ADDRESS))
    %,

    COPY_QUAD_ (FROM_QUAD, TO_QUAD) =
        BEGIN
```

```
        VECTOR [TO_QUAD, 0] = .VECTOR [FROM_QUAD, 0];
        VECTOR [TO_QUAD, 1] = .VECTOR [FROM_QUAD, 1];
        END
    %,

    DECREMENT_ (VALUE) =
        BEGIN
        VALUE = .VALUE - 1;
        END
    %,

    DESC_ADDR_ (DESC) =
        $BBLOCK [DESC, DSC$A_POINTER]
    %,

    DESC_CLASS_ (DESC) =
        $BBLOCK [DESC, DSC$B_CLASS]
    %,

    DESC_SIZE_ (DESC) =
        $BBLOCK [DESC, DSC$W_LENGTH]
    %,

    $DYNAMIC_DESC =
        $BBLOCK [8] PRESET (
        [DSC$W_LENGTH]      = 0,
        [DSC$B_DTYPE]       = DSC$K_DTYPE_T,
        [DSC$B_CLASS]       = DSC$K_CLASS_D,
        [DSC$A_POINTER]     = 0
            )
    %,

    FLINK_ (QUEUE_HEADER) =
        VECTOR [QUEUE_HEADER, 0]
    %,

    INCREMENT_ (VALUE) =
        BEGIN
        VALUE = .VALUE + 1
        END
    %,

    INIT_DYN_DESC_ (DESC) =
        BEGIN
        BIND X_DESC = DESC: $BBLOCK;
        X_DESC [DSC$W_LENGTH] = 0;
        X_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
        X_DESC [DSC$B_CLASS] = DSC$K_CLASS_D;
        X_DESC [DSC$A_POINTER] = 0;
        END
    %,

    INIT_QUEUE_HEADER_ (QUEUE_HEADER) =
        BEGIN
        FLINK_ (QUEUE_HEADER) = QUEUE_HEADER;
        BLINK_ (QUEUE_HEADER) = QUEUE_HEADER;
```

```
        END
    %,

    INIT_STAT_DESC_ (DESC, LENGTH, POINTER) =
        BEGIN
        BIND X_DESC = DESC: $BBLOCK;
        X_DESC [DSC$W_LENGTH] = LENGTH;
        X_DESC [DSC$B_DTYPE] = 0;
        X_DESC [DSC$B_CLASS] = 0;
        X_DESC [DSC$A_POINTER] = POINTER;
        END
    %,

    INSERT_HEAD_ (ENTRY_ADDR_, QUEUE_HEADER) =
        INSQUE (ENTRY_ADDR_, .FLINK_ (QUEUE_HEADER))
    %,

    INSERT_TAIL_ (ENTRY_ADDR_, QUEUE_HEADER) =
        INSQUE (ENTRY_ADDR_, .BLINK_ (QUEUE_HEADER))
    %,

    ITEM_PRESENT_ (ITEM_CODE) =
        BITVECTOR [SCB[PSM$Q_ITEM_FLAGS], %NAME ('SMBMSG$K_', ITEM_CODE)]
    %,

    OFFSET_TABLE_REPEAT_ [OFFSET, FIRST_BIT, SIZE, SIGN] =
        OFFSET
    %,

    PARAMETER_INDEX_ [] =
        BUILTIN NULLPARAMETER;
        LITERAL PARAMETER_INDEX_REPEAT_ (%REMAINING)
    %,

    PARAMETER_INDEX_REPEAT_ [PARAMETER] =
        %NAME ('_P_', PARAMETER) = %COUNT + 1
    %,

    PARAMETER_PRESENT_ (PARAM) =
        NOT NULLPARAMETER (%NAME ('_P_', PARAM))
    %,

    PRINT_FLAG_ (FLAG_NAME) =
        $BBLOCK [SCB[PSM$L_PRINT_FLAGS], %NAME ('SMBMSG$V_', FLAG_NAME)]
    %,

    READ_CHAR_ =
        BEGIN
        DECREMENT_ (SCB_SIZE_ (INPUT_RECORD));
        CH$RCHAR_A (SCB_ADDR_ (INPUT_RECORD))
        END
    %,

    REMOVE_HEAD_ (RESULT, QUEUE_HEADER) =
        REMQUE (.FLINK_ (QUEUE_HEADER), RESULT)
    %,
```

```
        REMOVE_TAIL_ (RESULT, QUEUE_HEADER) =
            REMQUE @.BLINK_ (QUEUE_HEADER), RESULT)
        %,

        REQUEST_FLAG_ (FLAG_NAME) =
            $BBLOCK [SCB[PSM$L_REQUEST_CONTROL], %NAME ('SMBMSG$V_', FLAG_NAME)]
        %,

        SEPARATE_FLAG_ (FLAG_NAME) =
            $BBLOCK [SCB[PSM$L_SEPARATION_CONTROL], %NAME ('SMBMSG$V_', FLAG_NAME)]
        %,

        SERVICE_LIST_ (SERVICE) =
            BITVECTOR [SCB[PSM$L_SERVICE_LIST], %NAME ('PSM$K_', SERVICE)]
        %,

        RETURN_IF_ERROR_ (ACTION) =
            BEGIN
            LOCAL STATUS;
            STATUS = ACTION;
            IF NOT .STATUS THEN RETURN (.STATUS);
            .STATUS
            END
        %,

        SET_DYN_DESC_ (DESC) =
            BEGIN
            $BBLOCK [SCB[%NAME ('PSM$Q_', DESC)], DSC$B_DTYPE] = DSC$K_DTYPE_T;
            $BBLOCK [SCB[%NAME ('PSM$Q_', DESC)], DSC$B_CLASS] = DSC$K_CLASS_D;
            END
        %,

        SIGNAL_IF_ERROR_ (ACTION) =
            BEGIN
            LOCAL STATUS;
            STATUS = ACTION;
            IF NOT .STATUS THEN SIGNAL (.STATUS);
            .STATUS
            END
        %,

        PSM$L_ = 0,0,32,0
        %,

        SCB_ADDR_ (DESC) =
            DESC_ADDR_ (SCB [ %NAME('PSM$Q_', DESC) ])
        %,

        SCB_CLASS_ (DESC) =
            DESC_CLASS_ (SCB [ %NAME('PSM$Q_', DESC) ])
        %,

        SCB_SIZE_ (DESC) =
            DESC_SIZE_ (SCB [ %NAME('PSM$Q_', DESC) ])
        %,
```

```
        WRITE_CHAR_ (CHAR) =
            BEGIN
            CH$WCHAR_A (CHAR, SCB_ADDR_ (OUTPUT_BUFFER));
            DECREMENT_ (SCB_SIZE_ (OUTPUT_BUFFER));
            END
        %
        ;
```

BANNER
LIS

PLIWRITE
LIS

SMBREQ
REQ

PLIVECTOR
LIS

SMBSRVSHR
MAP

FORMAT
LIS

PLIRODATA
LIS

SMBDEF
SDL

PRTSMB

DISPATCH
L13

PLISTRING
LIS

PRTSMB
MAP